On Undetectable Faults in Partial Scan Circuits Using Transparent-Scan

Irith Pomeranz[1]
School of Electrical & Computer Eng.
Purdue University
W. Lafayette, IN 47907

and

Sudhakar M. Reddy[2]
Electrical & Computer Eng. Dept.
University of Iowa
Iowa City, IA 52242

Abstract

We study the undetectable faults in partial scan circuits under a test application scheme referred to as transparent-scan. The transparent-scan approach allows very aggressive test compaction compared to other approaches. We demonstrate that, unlike other approaches that provide high levels of test compaction for partial scan circuits, this approach does not increase the number of undetectable faults. We also discuss the monotonicity of the number of undetectable faults with increased levels of scan.

1. Introduction

Test generation for scan circuits can be done under one of several test application schemes. Each approach has certain advantages that may make it the most suitable approach for a given circuit. Each approach also has its own set of undetectable faults. The study of undetectable faults is important in improving the test generation process for several reasons.

(1) A rigorous definition of undetectable faults and procedures to identify undetectable faults are important in defining an accurate fault efficiency metric. Such a metric is important in evaluating the effectiveness of a test generation procedure.

(2) Fast identification of undetectable faults alleviates the need to apply the test generation procedure to faults that cannot be detected. This can reduce the overall test generation time since a test generation procedure may spend large amounts of time considering undetectable faults.

Next, we describe each one of the three approaches to test generation for scan circuits and discuss its undetectable faults.

The first approach was considered for full scan circuits in [1]-[3] and for partial scan circuits in [4]. We refer to it as the *scan−per−vector* (*SPV*) approach. Under this approach, the present state variables (the outputs of the scan flip-flops) are considered as primary inputs of the circuit and the next state variables (the inputs of the scan flip-flops) are considered as primary outputs of the circuit during test generation. For a full scan circuit, combinational test generation is then applied to detect faults in the combinational logic of the circuit. The set of undetectable faults is the set of combinationally redundant faults. For a partial scan circuit, sequential test generation is required for a simplified circuit with a reduced number of state variables and increased numbers of primary inputs and primary outputs. The set of undetectable faults can be obtained using techniques developed for sequential circuits [5]-[10]. These techniques should be applied to the simplified circuit obtained after removing the scanned flip-flops and considering their outputs as inputs of the circuit and their inputs as outputs of the circuit. The test application scheme under this approach requires a scan-in operation before every primary input vector is applied and a scan-out operation after every primary input vector is applied. The state of the unscanned state variables should be held during a scan operation (other variations of this approach exist, for example, the values of the unscanned state variables can be obtained through the functional path of the circuit during scan; however, this complicates the test generation process).

The second approach was considered in [11]-[13]. We refer to it as the *scan−per−test* approach (*SPT*). Under this approach, the test generation procedure repeatedly selects between two options. The first option is to scan-out the current state and scan-in a new state. The second option is to continue applying primary input vectors without using the scan chain. The choice in [11]-[12] is made so as to minimize the number of clock cycles it will take to detect a target fault. The result is scan-based tests where sequences of primary input vectors are embedded between scan operations. Compared to *SPV*, the test application time for tests generated under *SPT* is lower since fewer scan operations are required (*SPV* requires a scan operation before and after every primary input vector). In addition, *SPT* allows testing of the circuit through its functional path for several clock cycles at a time. The set of undetectable faults under this approach was studied in [14] under the assumption that the values of the unscanned state variables are unknown after every scan-in operation. It was shown that for full scan circuits, the set of undetectable faults is the set of combinationally redundant faults. For partial scan circuits, a detectable fault under *SPV* may be undetectable under *SPT*. As a result, the set of undetectable faults under *SPT* is a superset of the set of undetectable faults under *SPV*.

Both *SPV* and *SPT* keep a clear distinction between scan operations and application of primary input vectors. Under *SPV*, each primary input vector is preceded and followed by a scan operation. Under *SPT*, sequences of primary input vectors are preceded and followed by scan operations. The third approach to testing of a scan circuit was considered in [15]. We refer to it as *transparent−scan* (*TRS*). Under this approach, the distinction between scan operations and application of primary input vectors is eliminated. Assuming a single scan chain, the test generation process explicitly uses extra inputs *scan_sel*, which is the select input of the scan multiplexers, and *scan_inp*, which is the input of the (single) scan chain. It also uses explicitly an extra output, *scan_out*, which is the output of the scan chain. When the test generation procedure sets $scan\_sel = 1$, a single shift of the scan chain results. If $scan\_sel = 1$ for N_{SH} consecutive time units, a *limited* scan operation that shifts the scan chain by N_{SH} positions results. As a consequence, limited scan operations are naturally incorporated into the test sequences generated by a sequential test generation procedure. This leads to very aggressive test compaction, as demonstrated in [15]. The transparent-scan approach was also found to be necessary in [16] for testing of critical paths in a microprocessor that uses partial scan.

The set of undetectable faults under *TRS* has not been studied before. We report on such a study here. Our results include the following.

1. Research supported in part by NSF Grant No. CCR-0098091 and in part by SRC Grant No. 2001-TJ-950.

2. Research supported in part by NSF Grant No. CCR-0097905 and in part by SRC Grant No. 2001-TJ-949.

(1) Denoting the set of undetectable faults under approach A (where $A = SPV$, SPT or TRS) by $F_{und,A}$, we prove that $F_{und,TRS} \subseteq F_{und,SPT}$. We also compare the numbers of undetectable faults under the three approaches in benchmark circuits. The comparison shows that the numbers of undetectable faults under SPV and under TRS are almost identical. Thus, while SPT achieves compaction at the cost of an increased number of undetectable faults, TRS achieves compaction without increasing the number of undetectable faults compared to SPV.

(2) We denote the set of undetectable faults under approach A in a circuit with a given set of scanned state variables S by $F^S_{und,A}$. For SPV and SPT, if $S_1 \subseteq S_2$, then $F^{S_1}_{und,A} \supseteq F^{S_2}_{und,A}$. For TRS we demonstrate that this may not always be the case. Thus, the set of undetectable faults under TRS needs to be computed for every scan configuration separately.

For ease of presentation we consider circuits with single scan chains. However, the results apply to circuits with multiple scan chains. The paper is organized as follows. In Section 2 we discuss the test application scheme under TRS. In Section 3 we consider the relationships between $F_{und,SPV}$, $F_{und,SPT}$ and $F_{und,TRS}$.

2. Partial scan under *TRS*

We first consider a full-scan circuit. If $scan\_sel = 1$ at time unit u, the state at time unit $u+1$ is obtained by shifting the state at time unit u by one position. We always shift states to the right. The leftmost state bit at time unit $u+1$ is equal to the value of $scan\_inp$ at time unit u. The value of $scan\_out$ is always equal to the value of the last flip-flop in the scan chain.

To simplify the discussion of partial scan, we always assume that when k out of K state variables are scanned, the scanned state variables are $y_0, \cdots, y_{k-1}$ and the unscanned state variables are $y_k, \cdots, y_{K-1}$. With this configuration, the following values are obtained at time unit $u+1$ if $scan\_sel = 1$ at time unit u. The values of $y_0, \cdots, y_{k-1}$ are determined by shifting the scan chain by one position to the right and setting y_0 equal to $scan\_inp$ at time unit u. The values of $y_k, \cdots, y_{K-1}$ are determined through the functional path of the circuit. More accurately, suppose that $y_i = \alpha_i$ at time unit u and that $scan\_inp = \alpha_{inp}$ at time unit u. Suppose in addition that if $scan\_sel = 0$ (i.e., the circuit operates in its functional mode at time unit u), $y_i = \beta_i$ at time unit $u+1$. Then the state at time unit $u+1$ has $y_0 = \alpha_{inp}$, $y_i = \alpha_{i-1}$ for $1 \leq i < k$, and $y_i = \beta_i$ for $k \leq i < K$.

3. Sets of undetectable faults

For non-scan synchronous sequential circuits, a definition of undetectable faults was given in [7]. This definition is the basis for defining undetectable faults in scan circuits under the various approaches we consider.

The definition of an undetectable fault in [7] is as follows. A fault f is detectable if there exists a test sequence T such that for every pair of initial states s of the fault free circuit and s' of the faulty circuit, the output sequence of the fault free circuit when its initial state is s and its input sequence is T is different from the output sequence of the faulty circuit when its initial state is s' and its input sequence is T. A fault f is undetectable if it is not detectable.

To identify undetectable faults under the SPV approach, the definition of [7] can be applied to the circuit obtained by removing the scanned flip-flops and considering their outputs as inputs of the circuit and their inputs as outputs of the circuit. With this modification, a scan operation is needed before and after every primary input vector. The scan-in operation assigns the necessary values to the scanned state variables and the scan-out operation observes their values one clock cycle later. The values of the unscanned state variables are left unchanged during a scan operation. This corresponds to our definition of SPV.

For the SPT approach, the definition of [7] has been modified in [14] to accommodate the fact that the state of the scanned state variables can be controlled at the beginning of a test and observed at its end, and that the values of the unscanned state variables are assumed to be unknown after the scan-in operation at the beginning of a test. Based on this definition, sufficient conditions for a fault to be undetectable were developed in [14]. A subset of the undetectable faults was found in [14] for benchmark circuits with different levels of scan based on these conditions.

To identify undetectable faults under the TRS approach, the definition of [7] can be applied directly to the scan circuit with the scan select, scan input, scan output and scan multiplexers inserted into the circuit.

Next, we show that if a fault f is detectable under SPT then the fault is also detectable under TRS. This will imply that $F_{und,TRS} \subseteq F_{und,SPT}$. In a similar way it can be shown that f is also detectable under SPV.

To show that if f is detectable under SPT then it is also detectable under TRS, we consider a test for f under SPT. We translate the test into a test under TRS. The existence of a test under TRS implies that f is detectable under TRS.

Consider a fault f which is detectable under SPT. A test under SPT has the form (SI, T), where SI is a scan-in vector and T is a sequence of primary input vectors. The test is applied as follows. First, SI is scanned in. At the end of the scan-in operation, the states of the unscanned flip-flops are assumed to be unknown. The primary input sequence is then applied. Finally, the state of the scanned state variables is scanned out. For f, let $SI = \alpha_0 \cdots \alpha_{k-1}$ and let $T = (t_0, \cdots, t_{L-1})$. We can apply this test under TRS as follows. For a time unit u such that $0 \leq u < k$, we use an arbitrary primary input vector, and we set $scan\_sel = 1$ and $scan\_inp = \alpha_{k-1-u}$. This brings the scanned state variables to state SI at time unit k. For a time unit u such that $k \leq u < k+L$, we apply the primary input vector t_{u-k} and we set $scan\_sel = 0$. The value of $scan\_inp$ is set arbitrarily. This applies the primary input sequence T to the circuit. For a time unit u such that $k+L \leq u < 2k+L$, we use an arbitrary primary input vector, we set $scan\_sel = 1$ and we assign an arbitrary value to $scan\_inp$. This results in scanning out of the final state of the scanned state variables. We obtain a test under TRS, which affects the circuit in exactly the same way as the SPT test (SI, T). Therefore, it detects the same fault f.

The relationship between $F_{und,TRS}$ and $F_{und,SPV}$ is not as simple for the following reason. Under SPV, the state of the unscanned state variables is held during a scan operation. Under TRS, the state of the unscanned state variables is determined through the functional path of the circuit during a scan operation. This difference can cause a fault to be detectable under one approach but undetectable under the other approach. It is also possible to consider a variation of SPV where the state of the unscanned state variables is determined through the functional path of the circuit during a scan operation. In this case, it is possible to translate a test under SPV into a test under TRS similar to the translation we performed from SPT to TRS. For this variation of SPV, the set of undetectable faults contains the set of undetectable faults under TRS.

In Tables 1, 2 and 3 we compare the numbers of undetectable faults under SPV, SPT and TRS experimentally. We use MCNC finite-state machine benchmarks. For SPV and TRS, we

implemented a procedure for identifying undetectable faults using the definition of [7], and we apply it to the appropriate circuits. For *SPT*, we repeat the results from [14], which provide a lower bound on the number of undetectable faults.

Table 1: Undetectable faults under *SPV*

		k=0		und					
circuit	flts	det	und	k=1	k=2	k=3	k=4	k=5	k=6
bbtas	63	62	1	0	0	0			
bbara	138	130	8	0	0	0	0		
bbsse	238	235	3	0	0	0	0		
dk512	124	122	2	0	0	0	0		
ex4	176	171	5	4	0	0	0		
ex7	160	149	11	10	1	1	1		
mark1	204	197	7	1	1	1	1		
opus	181	180	1	0	0	0	0		
train11	104	100	4	3	0	0	0		
dk16	532	529	3	2	2	2	2	2	
fetch	345	335	10	3	3	3	3	3	
keyb	470	468	2	0	0	0	0	0	
rie	552	545	7	4	4	4	4	4	
dvram	425	424	1	0	0	0	0	0	0

Table 2: Undetectable faults under *SPT* [14]

		k=0		und					
circuit	flts	det	und	k=1	k=2	k=3	k=4	k=5	k=6
bbtas	63	62	1	1	0	0			
bbara	138	130	8	8	7	1	0		
bbsse	238	235	3	2	2	1	0		
dk512	124	122	2	2	2	1	0		
ex4	176	171	5	5	5	5	0		
ex7	160	149	11	11	11	11	1		
mark1	204	197	7	7	7	4	1		
opus	181	180	1	0	0	0	0		
train11	104	100	4	4	2	0	0		
dk16	532	529	3	3	3	3	3	2	
fetch	345	335	10	8	8	8	7	3	
keyb	470	468	2	2	2	2	2	0	
rie	552	545	7	7	7	7	7	4	
dvram	425	424	1	1	1	1	1	1	0

Table 3: Undetectable faults under *TRS*

		k=0		und					
circuit	flts	det	und	k=1	k=2	k=3	k=4	k=5	k=6
bbtas	63	62	1	0	0	0			
bbara	138	130	8	0	0	0	0		
bbsse	238	235	3	0	0	0	0		
dk512	124	122	2	0	1	0	0		
ex4	176	171	5	4	0	0	0		
ex7	160	149	11	10	1	1	1		
mark1	204	197	7	1	4	1	1		
opus	181	180	1	0	0	0	0		
train11	104	100	4	3	0	0	0		
dk16	532	529	3	2	2	2	2	2	
fetch	345	335	10	3	4	3	3	3	
keyb	470	468	2	0	0	0	0	0	
rie	552	545	7	4	4	4	4	4	
dvram	425	424	1	0	0	0	0	0	0

Tables 1, 2 and 3 are organized as follows. After the circuit name we show the total number of collapsed single stuck-at faults. Under column $k = 0$ we consider the circuit without scan and we show the number of detectable faults and the number of undetectable faults in this circuit. Under column *und* subcolumn $k = i$ we show the number of undetectable faults in the scan circuit where the first k state variables are scanned. For *SPT*, this is a lower bound on the number of undetectable faults. We consider $0 \le k \le K$ where K is the number of state variables. The circuits are organized in groups with increasing numbers of state variables. The following points can be seen from Tables 1, 2 and 3.

Both *SPV* and *TRS* have significantly fewer undetectable faults than *SPT*. Even with low levels of scan ($k = 1$), the set of undetectable faults under *SPV* and *TRS* typically includes only the combinationally redundant faults (which are also undetectable when $k = K$). The numbers of undetectable faults under *SPV* and *TRS* are almost identical. This indicates that the advantages of *TRS* (high levels of test compaction and the option for functional testing) are obtained without loss in the achievable fault coverage.

Considering *dk*512, *mark*1 and *fetch* under *TRS* it can be seen that the number of undetectable faults does not necessarily go down as the number of scanned state variables is increased. This does not occur for *SPV* and *SPT*, where the set of undetectable faults decreases monotonically as additional flip-flops are added to the scan chain.

4. Concluding remarks

We studied undetectable faults in scan circuits under a test application scheme referred to as *transparent−scan* (or *TRS*). Our study yielded the following results in comparison to two other approaches, *SPV* and *SPT*. (1) The numbers of undetectable faults under *SPV* and under *TRS* are almost identical. Thus, while *SPT* achieves compaction at the cost of an increased number of undetectable faults, *TRS* achieves even higher levels of compaction without increasing the number of undetectable faults compared to *SPV*. (2) The set of undetectable faults under *TRS* does not decrease monotonically as the level of scan is increased, and needs to be computed for every scan configuration separately.

References

[1] P. Goel, "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits", IEEE TC, March 1981, pp. 215-222.

[2] H. Fujiwara and T. Shimono, "On the Acceleration of Test Generation Algorithms", IEEE TC, Dec. 1983, pp. 1137-1144.

[3] M. Schulz et al., "SOCRATES: A Highly Efficient Automatic Test Pattern Generation System", IEEE TCAD, Jan. 1988, pp. 126-137.

[4] X. Lin, I. Pomeranz and S. M. Reddy, "Full Scan Fault Coverage With Partial Scan", in Proc. DATE-99, pp. 468-472.

[5] V. D. Agrawal and S. T. Chakradhar, "Combinational ATPG Theorems for Identifying Untestable Faults in Sequential Circuits", in Proc. ETC-93, pp. 249-253.

[6] K.-T. Cheng, "On Removing Redundancy in Sequential Circuits", in Proc. of DAC-91, pp. 164-169.

[7] I. Pomeranz and S. M. Reddy, "Classification of Faults in Synchronous Sequential Circuits", IEEE TC, Sept., 1993, pp. 1066-1077.

[8] M. A. Iyer and M. Abramovici, "Sequentially Untestable Faults Identified Without Search (Simple Implications Beat Exhaustive Search!)", in Proc. ITC-94, pp. 259-266.

[9] X. Lin, I. Pomeranz and S. M. Reddy, "On Finding Undetectable and Redundant Faults in Synchronous Sequential Circuits", in Proc. ICCD-98, pp. 498-503.

[10] Q. Peng, M. Abramovici and J. Savir, "MUST: Multiple-Stem Analysis for Identifying Sequentially Untestable Faults", in Proc. ITC-00, pp. 839-846.

[11] D. K. Pradhan and J. Saxena, "A Design for Testability Scheme to Reduce Test Application Time in Full Scan", in Proc. VTS-92, pp. 55-60.

[12] S. Y. Lee and K. K. Saluja, "Test Application Time Reduction for Sequential Circuits with Scan", IEEE TCAD, Sept. 1995, pp. 1128-1140.

[13] I. Pomeranz and S. M. Reddy, "Simulation Based Test Generation for Scan Designs", in Proc. ICCAD-00, pp. 544-549.

[14] I. Pomeranz and S. M. Reddy, "On Undetectable Faults in Partial Scan Circuits", in Proc. ICCAD-02, pp. 82-86.

[15] I. Pomeranz and S. M. Reddy, "A New Approach to Test Generation and Test Compaction for Scan Circuits", in Proc. DATE-03, pp. 1000-1005.

[16] J. Grodstein, D. Bhavsar, V. Bettada and R. Davies, "Automatic Generation of Critical-Path Tests for a Partial-Scan Microprocessor", in Proc. ICCD, Oct. 2003.